# Create an Azure AD Application for App-Owns-Data Embedding

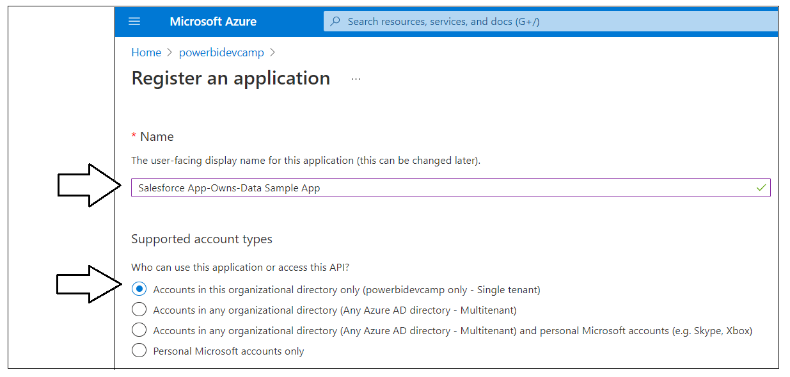
Follow the steps in this document to create a new Azure AD application for the [**SalesforceAppOwnsDataEmbedding**](https://github.com/PowerBiDevCamp/SalesforceAppOwnsDataEmbedding/tree/main/SalesforceAppOwnsDataEmbedding/force-app/main/default/aura/powerBiReportAura) sample project. To complete these steps, you will require a Power BI environment in which you have a user account that has been configured as a Power BI Service admin. If you do not have a Power BI environment for testing, you can create one for free by following the steps in [Create a Development Environment for Power BI Embedding](https://github.com/PowerBiDevCamp/Camp-Sessions/blob/master/Create%20Power%20BI%20Development%20Environment.pdf).

When you login to the Azure portal to create the new Azure AD application, make sure you log in using a user account in the same tenant which contains the Power BI reports you'd like to embed. Begin by navigating to the [App registration](https://portal.azure.com/#blade/Microsoft_AAD_IAM/ActiveDirectoryMenuBlade/RegisteredApps) page in the Azure portal and click the **New registration** link.

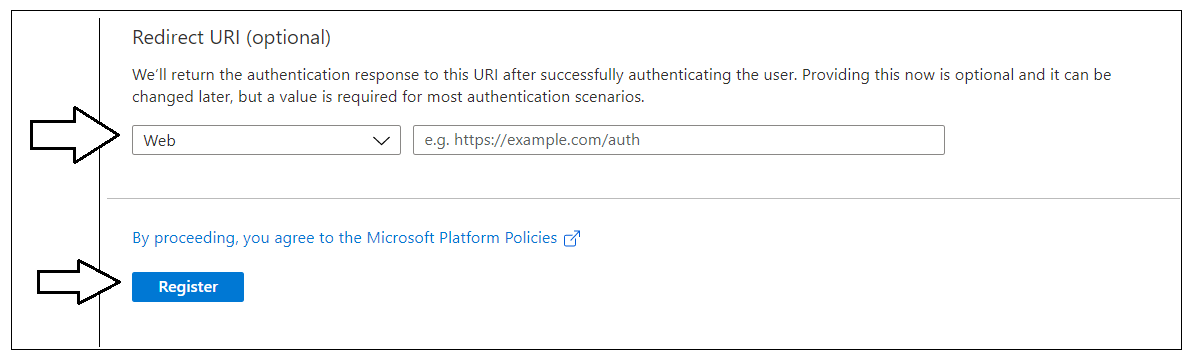
Graphical user interface, text, application, email

Description automatically generated

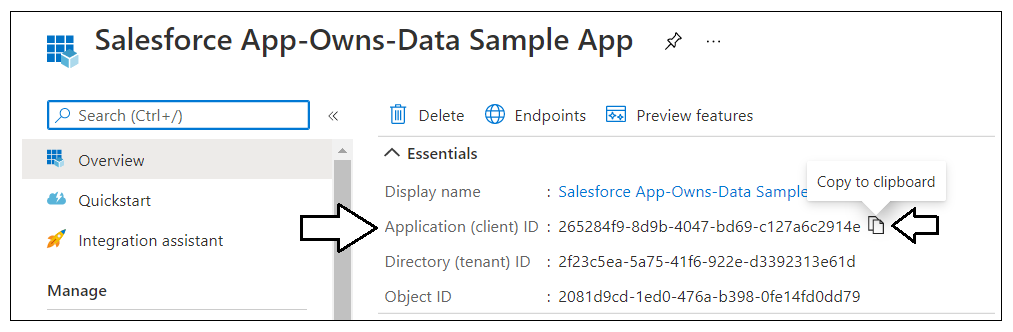
On the **Register an application** page, enter an application name such as **Salesforce App-Owns-Data Sample App** and accept the default selection for **Supported account types** of **Accounts in this organizational directory only**.



In the **Redirect URI** section leave the default selection in the dropdown box to **Web** and leave the textbox to the right of the dropdown box empty. The reason for leaving this textbox empty is that you do not need to add a Redirect URI when acquiring app-only access tokens using Client Credentials Flow. Click the **Register** button to create the new Azure AD application.



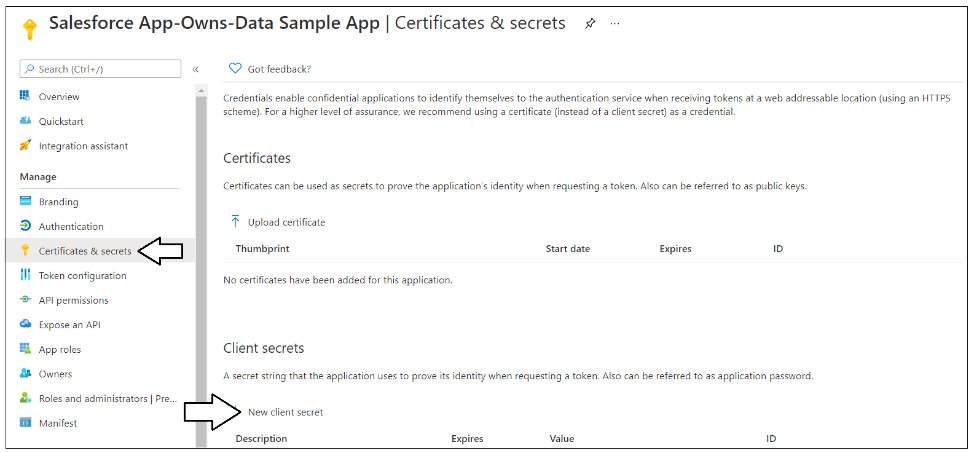
After creating a new Azure AD application in the Azure portal, you should see the Azure AD application overview page which displays the **Application ID**. Note that the *Application ID* is often called the *Client ID*. You will need to copy this Application ID and use it later to configure the project's support for Client Credentials Flow.



Copy the Client ID (aka Application ID) and paste it into a text document so you can use it later in the setup process. Note that this is the Client ID value that will be used by **SalesforceAppOwnsDataEmbedding** project to authenticate as a service principal.



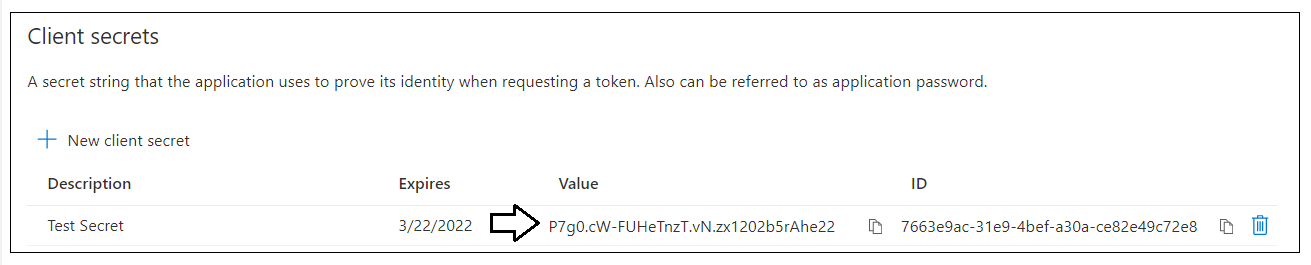
Next, you need to create a Client Secret for the application. The Client Secret will act as the application's password when it authenticates using Client Credentials Flow. Click on the **Certificates & secrets** link in the left navigation to move to the **Certificates & secrets** page. On the **Certificates & secrets** page, click the **New client secret** button as shown in the following screenshot.



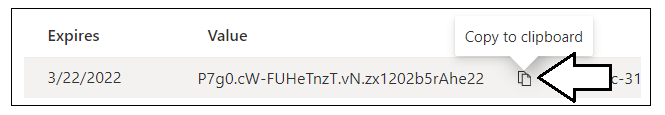
In the **Add a client secret** dialog, add a text description such as **Test Secret** and then click the **Add** button to create the new Client Secret.



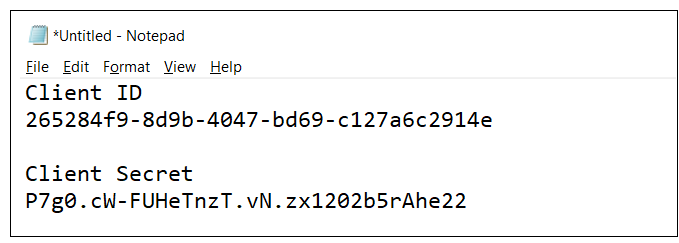
Once you have created the Client Secret, you should be able to see its **Value** in the **Client secrets** section.



Click on the **Copy to clipboard** button to copy the Client Secret into the clipboard.



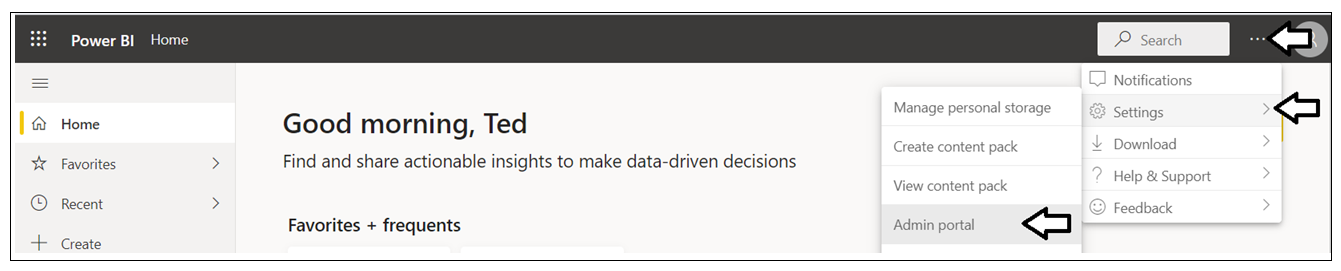
Paste the Client Secret into the same text document with the Client ID.



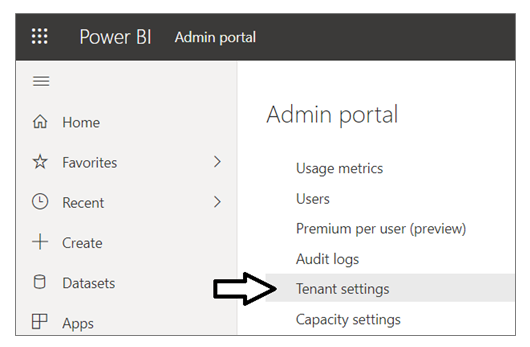
Later in the setup process, you will need to configure the **SalesforceAppOwnsDataEmbedding** project with these values for the Client ID and the Client Secret. Note that these credentials values will be tracked in the Salesforce environment using a Custom Metadata Type.

## Configure Service Principal Support in the Power BI Tenant

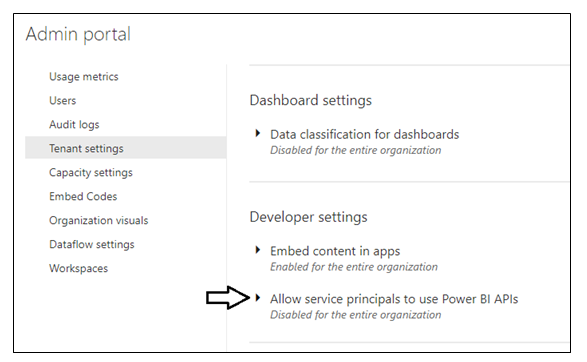
Enable the Allow service principals to use Power BI APIs setting and configure it with the Power BI Apps security group. Navigate to the Power BI portal at <https://app.powerbi.com>. Drop down the Settings menu and select the navigation command for the Admin portal.



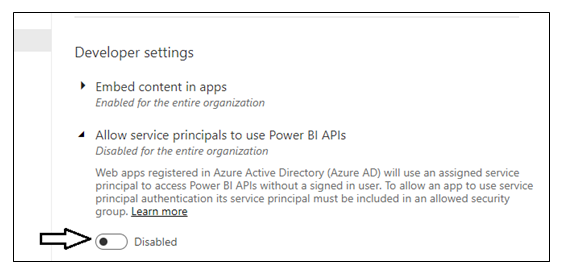
In the Power BI Admin portal, click the **Tenant settings** link on the left.



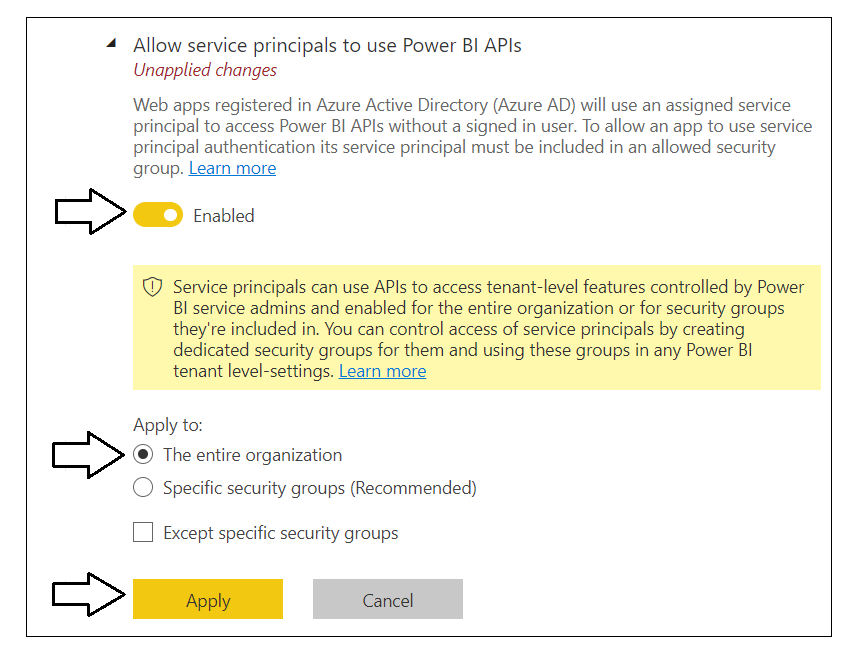
Move down in the **Developer settings** section and expand the **Allow service principals to use Power BI APIs** section.



Note that the Allow service principals to use Power BI APIs setting is initially set to Disabled.



Change the setting to **Enabled** and set the **Apply to** setting to **The entire organization**. Click the **Apply** button to configure the support you need for a service principal to call the Power BI Service API.



You will see a notification indicating it may take up to 15 minutes until your tenant recognizes your configuration changes.

Text

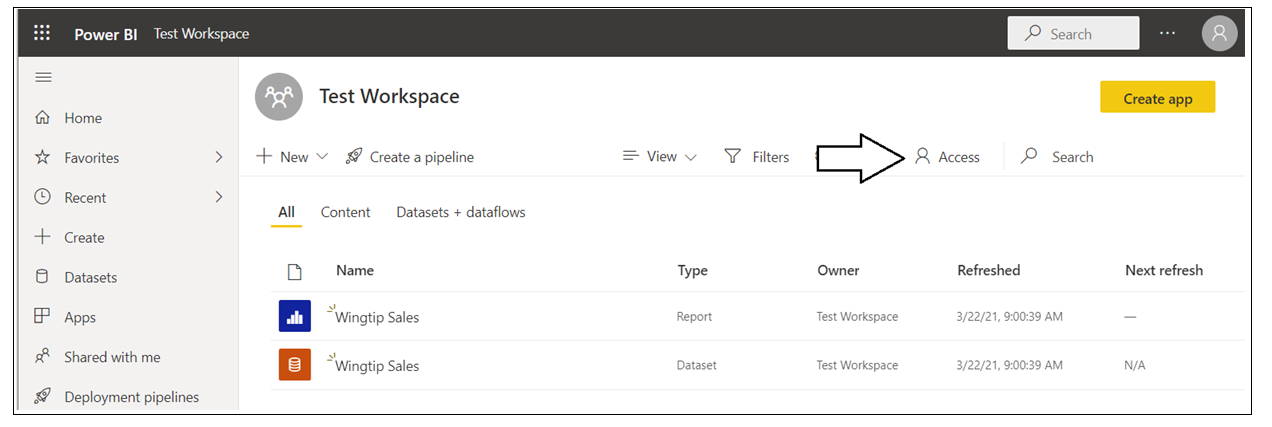
Description automatically generated with medium confidence

## Configure Service Principal Access to a Power BI Workspace

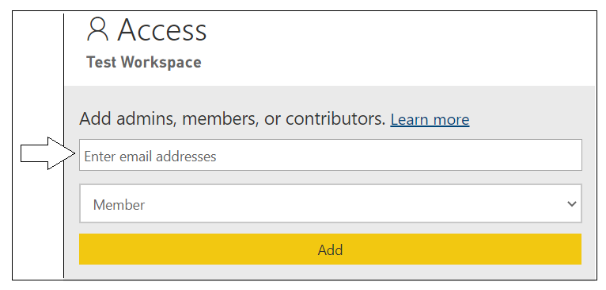
Add the Azure AD service principal for the Azure AD application to a Power BI workspace as an administrator.

Click the Dev Camp Lab workspace in the left navigation to display the workspace summary page.

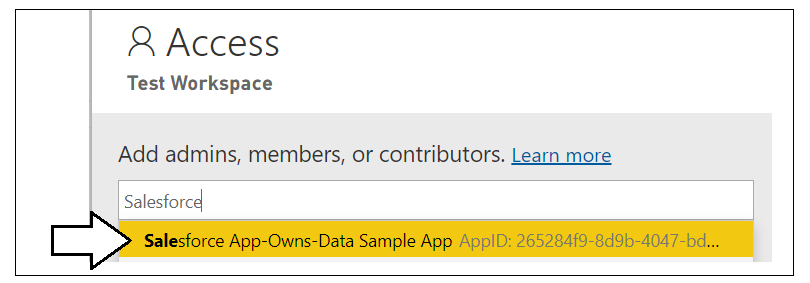
Click the Access link to open the Access pane where you can configure who has access to workspace resources.



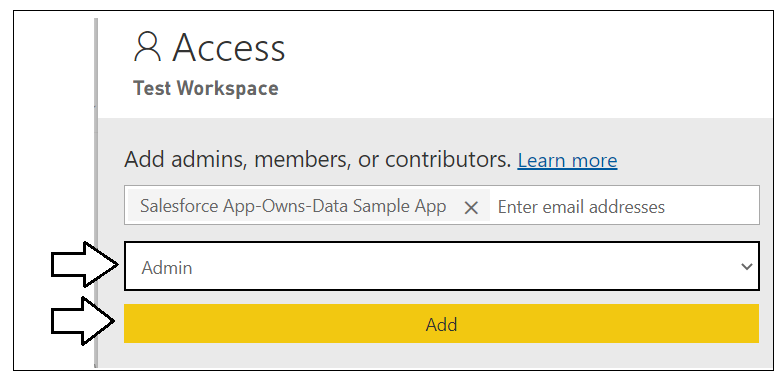
In the search box with the caption of Enter email address, type App-Owns-Data to find the Azure AD application.



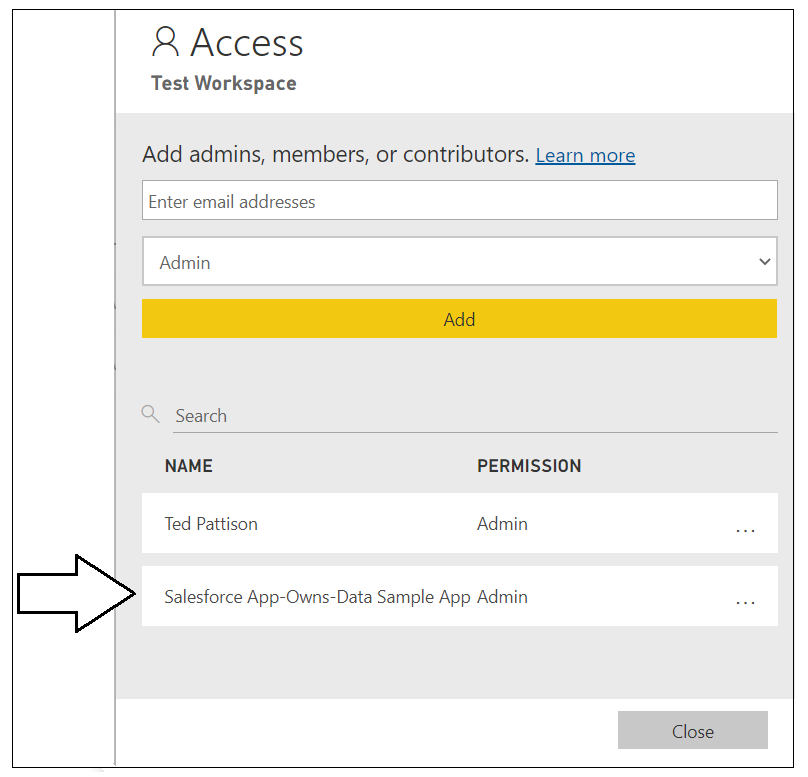
In the search box with the caption of Enter email address, type App-Owns-Data to find the Azure AD application.



Select the Azure AD application named App-Owns-Data Sample App. Select Admin in the dropdown menu to specify the level of access and then click the Add button.



You should now be able to confirm that the App-Owns-Data Sample App has been configured as a workspace admin.



Close the Access pane.

Graphical user interface, text, application, email

Description automatically generated